# Problem 4: Water Bottles 4+3=7 Point(s)

Problem ID: bottles

Rank: 2+3

### Introduction

Berkeley students living in some parts of the Foothill residential complex source most potable water from a communal water dispenser. This problem is inspired by an everyday technique we use to reduce the awkwardness of delaying people behind us in line! The real dispenser outputs 2 liters per minute, but we'll say 1 per minute for this problem to make things simpler.

## **Problem Statement**

**N** students numbered 1, ..., **N** are lined up at a water dispenser that dispenses water at a constant rate of 1 liter per minute. The  $i^{th}$  student has an empty bottle with capacity  $\mathbf{C}_i$  liters that they begin to fill immediately after the previous student has finished (formally, the  $i^{th}$  student begins refilling when all  $j^{th}$  students for which j < i finish refilling).

We define the *wait time* of a student as the **total** time they have to wait until their bottle **finishes** refilling. The students ask you to reorder the line into a new permutation  $(a_1, ..., a_N)$  of the students' numbers such that the students' total wait time is minimized.

For the bonus test set only, the students also require a tiebreaker: If multiple permutations result in the minimum total wait time, choose any permutation that minimizes the number of students moved to a new position. Formally, minimize the number of indices i for which  $a_i \neq i$ . If there are multiple permutations that accomplish this, you may choose any.

# **Input Format**

The first line of the input contains an integer **T**, denoting the number of test cases that follow. For each test case:

- The first line contains a positive integer N denoting the number of students in line.
- The second line contains a sequence of N positive integers  $C_1$ , ...,  $C_N$ , denoting the bottle capacities in liters.

# **Output Format**

For each test case, output the following two lines:

- On the first line, output the minimum total wait time in minutes.
- On the second line, output N integers a<sub>1</sub>, ..., a<sub>N</sub> (1 ≤ a<sub>i</sub> ≤ N) where a<sub>i</sub> is the new index of the i<sup>th</sup> student from the front of the original line. If there are multiple permutations that satisfy all criteria, you may output any.

## **Constraints**

 $1 \le T \le 100$ 

#### **Main Test Set**

 $1 \le \mathbf{N} \le 10^3$ 

The sum of N across all test cases in a test file does not exceed  $10^3$ .

 $1 \le \mathbf{C}_i \le 10^3$ 

All capacities  $C_i$  are distinct.

There is guaranteed to exist exactly one optimal permutation for each test case.

#### **Bonus Test Set**

 $1 \le N \le 10^5$ 

The sum of N across all test cases in a test file does not exceed 10<sup>5</sup>.

 $1 \le \mathbf{C}_i \le 10^9$ 

The capacities  $C_i$  are not guaranteed to be distinct.

## **Sample Test Cases**

Main Sample Input	<u>Download</u>	Main Sample Output	<u>Download</u>
1		12	
3		2 3 1	
5 1 2			

#### **Main Sample Explanations**

The optimal permutation rearranges the line into the order (2, 3, 1).

- 1. Student 2 is first in the new line. They have a 1 L bottle and spend 1 minute refilling it.
- 2. Student 3 is second. They have to wait 1 minute and then spend 2 minutes refilling their own bottle, finishing after 3 minutes.
- 3. Student 1 is third. They have to wait 3 minutes and then spend 5 minutes refilling their own bottle, finishing after 8 minutes.

The total wait time is (1 + 3 + 8) min = 12 minutes, and it can be shown that no other permutation results in a total less than or equal to than 12 minutes.

Bonus Sample Input	<u>Download</u>	<b>Bonus Sample Output</b>	<u>Download</u>
2 3 2 1 1 4 2 2 1 1		7 3 2 1 13 3 4 1 2	
		Note that this is one of many possible correct outputs. If there are multiple solutions, you may output any of them.	

#### **Bonus Sample Explanations**

For test case #1, the permutations (2, 3, 1) and (3, 2, 1) both achieve the minimum total wait time of 7 minutes. However, (2, 3, 1) moves three students while (3, 2, 1) moves only two, so (3, 2, 1) is preferred.

For test case #2, the permutation (3, 4, 1, 2) moves all students and results in a total wait time of 13 minutes, which can be shown to be the minimum. There are also three other permutations that result in a total wait time of 13 minutes and move all students: (3, 4, 2, 1), (4, 3, 1, 2), and (4, 3, 2, 1). The tiebreaker is thus inconclusive—any of these permutations are acceptable.